

1) Build a Responsive Pricing Table

Use Flexbox or CSS Grid for layout.

Each plan should be in a card format with listed features.

Highlight the “Pro” plan with a different background and hover scale animation.

Use media queries to make it stack vertically on smaller screens.

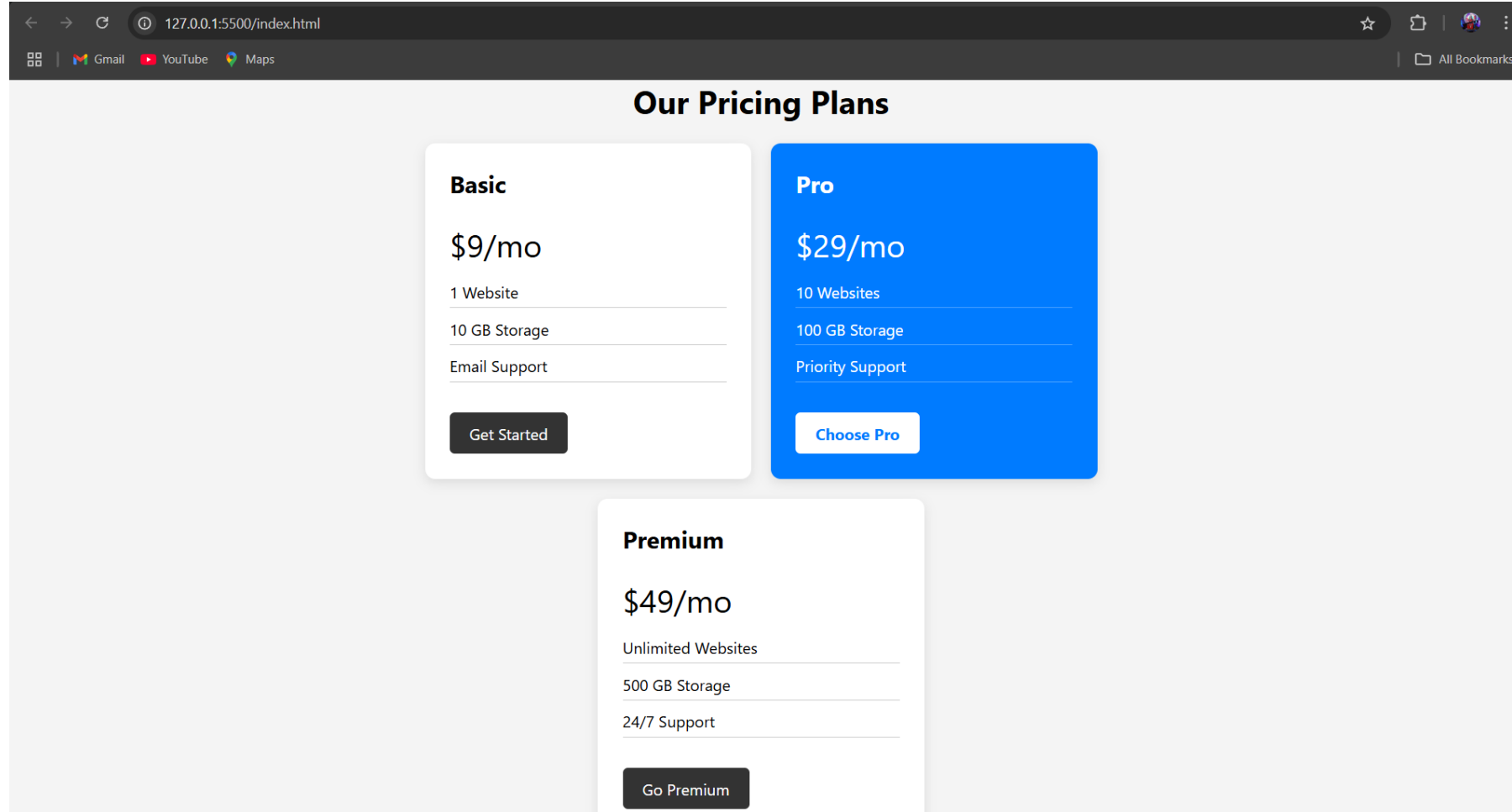
```
index.html X # style.css
index.html > html > head
2 <html lang="en">
3 </html>
4 </head>
5 <body>
6
7 <h1 style="text-align:center;">Our Pricing Plans</h1>
8
9
10 <div class="pricing-table">
11 <div class="card">
12 <h3>Basic</h3>
13 <div class="price">$9/mo</div>
14 <ul>
15 <li>1 Website</li>
16 <li>10 GB Storage</li>
17 <li>Email Support</li>
18 </ul>
19 <a href="#" class="btn">Get Started</a>
20 </div>
21
22 <div class="card pro">
23 <h3>Pro</h3>
24 <div class="price">$29/mo</div>
25 <ul>
26 <li>10 Websites</li>
27 <li>100 GB Storage</li>
28 <li>Priority Support</li>
29 </ul>
30 <a href="#" class="btn">Choose Pro</a>
31 </div>
32
33 <div class="card">
34 <h3>Premium</h3>
35 <div class="price">$49/mo</div>
36 <ul>
37 <li>Unlimited Websites</li>
38 <li>500 GB Storage</li>
39 <li>24/7 Support</li>
40 </ul>
41 <a href="#" class="btn">Go Premium</a>
42 </div>
43 </div>
44
45
```

```
index.html X # style.css X
# style.css > ...
1
2 body {
3   font-family: 'Segoe UI', sans-serif;
4   background-color: #f4f4f4;
5   margin: 0;
6   padding: 40px 20px;
7 }
8
9 .pricing-table {
10  display: flex;
11  gap: 20px;
12  justify-content: center;
13  flex-wrap: wrap;
14  max-width: 1000px;
15  margin: auto;
16 }
17
18 .card {
19  background-color: white;
20  border-radius: 10px;
21  padding: 25px;
22  width: 280px;
23  box-shadow: 0 4px 12px rgba(0,0,0,0.1);
24  transition: transform 0.3s ease;
25 }
26
27 .card:hover {
28  transform: scale(1.05);
29 }
30
31 .card.pro {
32  background-color: #007bff;
33  color: white;
34 }
35
36 .card h3 {
37  margin-top: 0;
38  font-size: 24px;
39 }
40
```

```
index.html # style.css X
# style.css > ...
36 .card h3 {
37 }
38
39 .price {
40   font-size: 32px;
41   margin: 10px 0;
42 }
43
44 ul {
45   list-style: none;
46   padding: 0;
47 }
48
49 ul li {
50   margin: 10px 0;
51   border-bottom: 1px solid #ccc;
52   padding-bottom: 5px;
53 }
54
55 .card.pro ul li {
56   border-color: rgba(255,255,255,0.3);
57 }
58
59 .btn {
60   display: inline-block;
61   margin-top: 15px;
62   padding: 10px 20px;
63   background-color: #333;
64   color: white;
65   text-decoration: none;
66   border-radius: 5px;
67 }
68
69 .card.pro .btn {
70   background-color: #fff;
71   color: #007bff;
72   font-weight: bold;
73 }
74
75
76
```

```
77 @media (max-width: 768px) {
78   .pricing-table {
79     flex-direction: column;
80     align-items: center;
81   }
82 }
83
```

1)OUTPUT



2) Create a CSS-Based Image Gallery

Use a grid layout to display images.

On hover, images should zoom and show a caption using ::after or ::before.

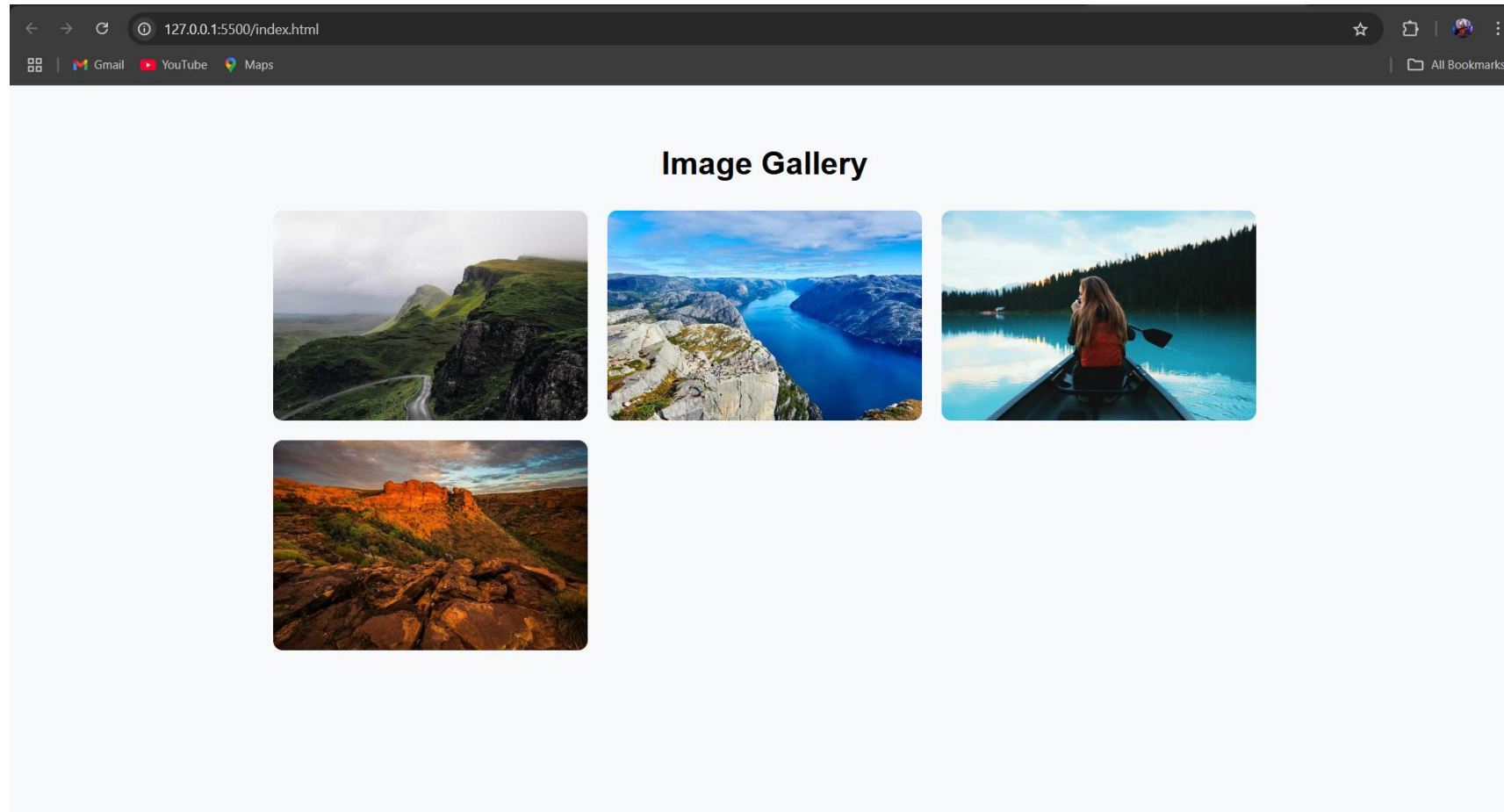
Adjust to single-column layout on mobile.

```
index.html > html > head > link
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>CSS Image Gallery</title>
6   <link rel="stylesheet" href="style.css">
7 </head>
8 <body>
9
10  <h1>Image Gallery</h1>
11
12  <div class="gallery">
13    <div class="gallery-item" data-caption="Beautiful Mountain">
14      
15    </div>
16    <div class="gallery-item" data-caption="Calm Lake">
17      
18    </div>
19    <div class="gallery-item" data-caption="Cityscape at Night">
20      
21    </div>
22    <div class="gallery-item" data-caption="Sunset Beach">
23      
24    </div>
25  </div>
26
27 </body>
28 </html>
29
```

```
index.html # style.css X
# style.css > ...
1
2 body {
3   font-family: Arial, sans-serif;
4   background: #f8f9fa;
5   margin: 0;
6   padding: 40px 20px;
7 }
8
9 h1 {
10  text-align: center;
11  margin-bottom: 30px;
12 }
13
14 .gallery {
15   display: grid;
16   grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
17   gap: 20px;
18   max-width: 1000px;
19   margin: auto;
20 }
21
22 .gallery-item {
23   position: relative;
24   overflow: hidden;
25   border-radius: 10px;
26   cursor: pointer;
27 }
28
29 .gallery-item img {
30   width: 100%;
31   height: auto;
32   display: block;
33   transition: transform 0.4s ease;
34 }
35
36 .gallery-item::after {
37   content: attr(data-caption);
38   position: absolute;
39   bottom: 0;
40   left: 0;
41 }
42
```

```
index.html # style.css X
# style.css > ...
36 .gallery-item::after {
37   bottom: 0;
38   left: 0;
39   right: 0;
40   background: rgba(0,0,0,0.6);
41   color: #fff;
42   padding: 10px;
43   font-size: 16px;
44   opacity: 0;
45   transform: translateY(100%);
46   transition: opacity 0.4s ease, transform 0.4s ease;
47   text-align: center;
48 }
49
50
51 .gallery-item:hover img {
52   transform: scale(1.1);
53 }
54
55 .gallery-item:hover::after {
56   opacity: 1;
57   transform: translateY(0);
58 }
59
60
61 @media (max-width: 600px) {
62   .gallery {
63     grid-template-columns: 1fr;
64   }
65 }
66
```

2) OUTPUT



3)Design a Themed Landing Page

Include navigation, hero, features, and footer sections.

Use CSS variables for consistent color and spacing.

Add scroll-based CSS transitions (like fade/slide in).

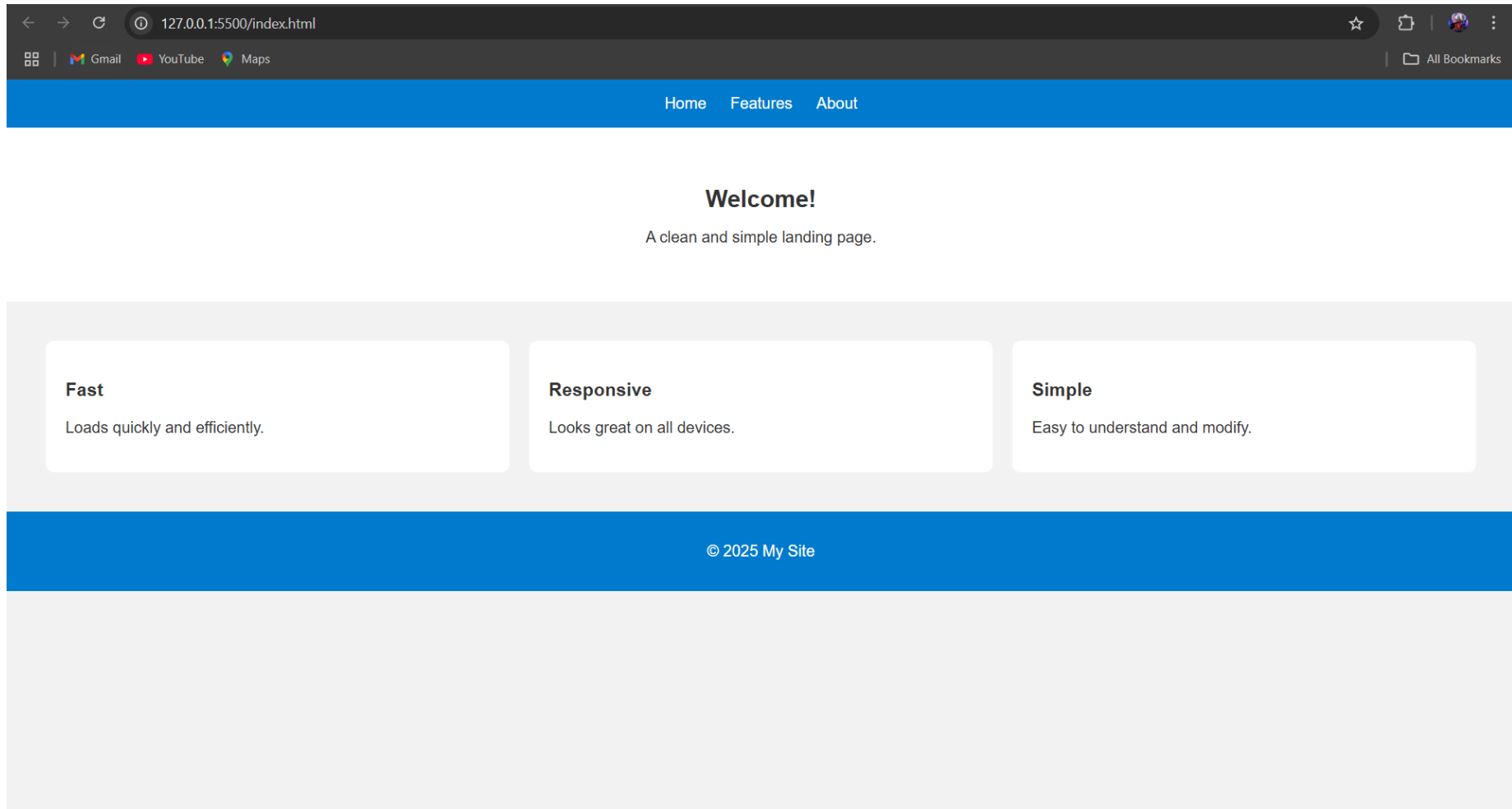
Make it fully responsive.

```
index.html X # style.css
index.html > html > head > link
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <title>Simple Landing Page</title>
7 <link rel="stylesheet" href="style.css">
8 </head>
9 <body>
10
11 <header>
12 <nav>
13 <a href="#">Home</a>
14 <a href="#">Features</a>
15 <a href="#">About</a>
16 </nav>
17 </header>
18
19 <section class="hero">
20 <h1>Welcome!</h1>
21 <p>A clean and simple landing page.</p>
22 </section>
23
24 <section class="features">
25 <div><h3>Fast</h3><p>Loads quickly and efficiently.</p></div>
26 <div><h3>Responsive</h3><p>Looks great on all devices.</p></div>
27 <div><h3>Simple</h3><p>Easy to understand and modify.</p></div>
28 </section>
29
30 <footer>
31 <p>&copy; 2025 My Site</p>
32 </footer>
33
34 </body>
35 </html>
36
37
```

```
index.html X # style.css X
# style.css > ...
1
2 :root {
3   --main: #007acc;
4   --bg: #f2f2f2;
5   --text: #333;
6   --pad: 40px;
7 }
8
9 body {
10  margin: 0;
11  font-family: sans-serif;
12  background: var(--bg);
13  color: var(--text);
14 }
15
16 header, footer {
17  background: var(--main);
18  color: white;
19  padding: 15px;
20  text-align: center;
21 }
22
23 nav a {
24  color: white;
25  margin: 0 10px;
26  text-decoration: none;
27 }
28
29 .hero {
30  padding: var(--pad);
31  text-align: center;
32  background: white;
33 }
34
35 .hero h1 {
36  margin-bottom: 10px;
37 }
38
39 .features {
40
```

```
index.html # style.css X
# style.css > ...
35 .hero h1 {
36  margin-bottom: 10px;
37 }
38
39 .features {
40  display: grid;
41  grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
42  gap: 20px;
43  padding: var(--pad);
44 }
45
46 .features div {
47  background: white;
48  padding: 20px;
49  border-radius: 8px;
50  opacity: 0;
51  transform: translateY(30px);
52  animation: fadeUp 1s forwards;
53 }
54
55 .features div:nth-child(2) { animation-delay: 0.2s; }
56 .features div:nth-child(3) { animation-delay: 0.4s; }
57
58 @keyframes fadeUp {
59  to {
60    opacity: 1;
61    transform: translateY(0);
62  }
63 }
64
65 @media (max-width: 600px) {
66  .hero h1 { font-size: 1.5rem; }
67 }
68
```

3)OUTPUT



4) Build a Vertical Timeline Component

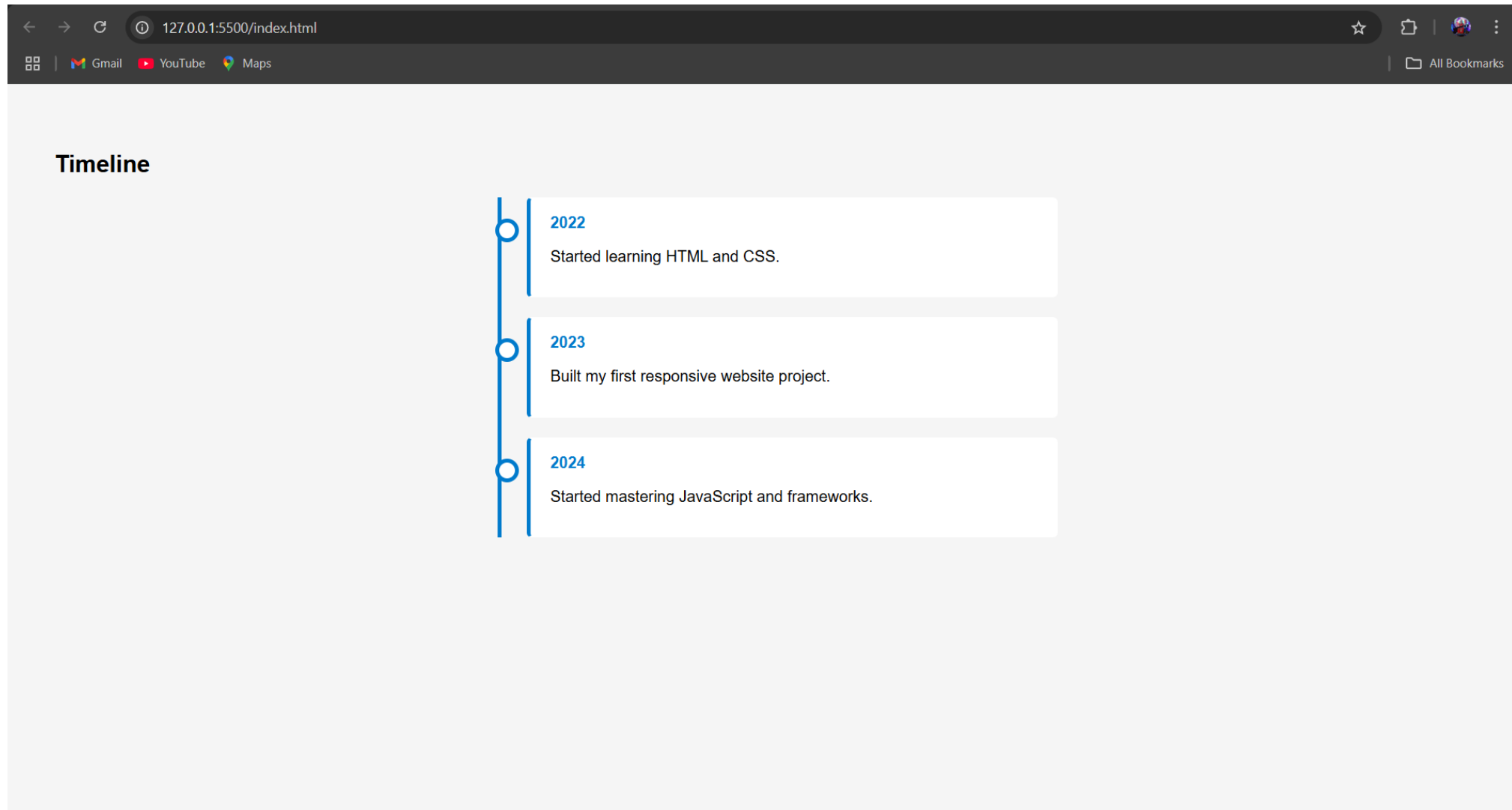
- Use pseudo-elements (::before, ::after) for connecting lines.
- Each event is a styled card with date and description.
- Hovering changes background or border style.

```
index.html X # style.css
C:\Users\HP\Desktop\Assignment_1\index.html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8" />
5   <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
6   <title>Vertical Timeline</title>
7   <link rel="stylesheet" href="style.css">
8
9 </head>
10 <body>
11
12   <h2>Timeline</h2>
13   <div class="timeline">
14     <div class="event">
15       <div class="date">2022</div>
16       <p>Started learning HTML and CSS.</p>
17     </div>
18     <div class="event">
19       <div class="date">2023</div>
20       <p>Built my first responsive website project.</p>
21     </div>
22     <div class="event">
23       <div class="date">2024</div>
24       <p>Started mastering JavaScript and frameworks.</p>
25     </div>
26   </div>
27
28 </body>
29 </html>
```

```
index.html X # style.css X
# style.css > ...
1
2   body {
3     font-family: sans-serif;
4     background: #f5f5f5;
5     padding: 40px;
6   }
7
8   .timeline {
9     position: relative;
10    margin: 0 auto;
11    max-width: 600px;
12  }
13
14  .timeline::before {
15    content: '';
16    position: absolute;
17    left: 30px;
18    top: 0;
19    bottom: 0;
20    width: 4px;
21    background: #007acc;
22  }
23
24  .event {
25    position: relative;
26    margin-left: 60px;
27    background: white;
28    border: 2px solid transparent;
29    border-left: 4px solid #007acc;
30    padding: 15px 20px;
31    margin-bottom: 20px;
32    border-radius: 5px;
33    transition: 0.3s ease;
34  }
35
36  .event::before {
37    content: '';
38    position: absolute;
39    left: -36px;
40    top: 20px;
```

```
index.html # style.css X
# style.css > ...
36 .event::before {
39   left: -36px;
40   top: 20px;
41   width: 16px;
42   height: 16px;
43   background: #fff;
44   border: 4px solid #007acc;
45   border-radius: 50%;
46 }
47
48 .event:hover {
49   background: #e0f4ff;
50   border-color: #005999;
51 }
52
53 .date {
54   font-weight: bold;
55   color: #007acc;
56   margin-bottom: 5px;
57 }
58
59 @media (max-width: 600px) {
60   .timeline::before {
61     left: 20px;
62   }
63
64   .event {
65     margin-left: 50px;
66   }
67
68   .event::before {
69     left: -26px;
70   }
71 }
72
```

4)OUTPUT



5) Implement a Tabbed Interface (Pure CSS)

Use radio buttons or checkboxes to switch tabs.

CSS selectors control tab content visibility.

Add transitions for a smooth UX.

```
index.html X # style.css
index.html > html > head
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>CSS Tabs</title>
7
8 </head>
9 <body>
10
11   <div class="tabs">
12     <input type="radio" id="tab1" name="tab" checked>
13     <label for="tab1">Tab 1</label>
14
15     <input type="radio" id="tab2" name="tab">
16     <label for="tab2">Tab 2</label>
17
18     <input type="radio" id="tab3" name="tab">
19     <label for="tab3">Tab 3</label>
20
21     <div id="content1" class="tab-content">
22       <h3>Tab 1 Content</h3>
23       <p>This is content for the first tab.</p>
24     </div>
25
26     <div id="content2" class="tab-content">
27       <h3>Tab 2 Content</h3>
28       <p>This is content for the second tab.</p>
29     </div>
30
31     <div id="content3" class="tab-content">
32       <h3>Tab 3 Content</h3>
33       <p>This is content for the third tab.</p>
34     </div>
35   </div>
36
37 </body>
38 </html>
39
```

```
index.html X # style.css X
# style.css > ...
1
2   body {
3     font-family: sans-serif;
4     background: #f0f0f0;
5     padding: 40px;
6   }
7
8   .tabs {
9     max-width: 500px;
10    margin: auto;
11  }
12
13  .tabs input[type="radio"] {
14    display: none;
15  }
16
17  .tabs label {
18    padding: 10px 20px;
19    display: inline-block;
20    background: #ddd;
21    cursor: pointer;
22    transition: 0.3s;
23    border-top-left-radius: 5px;
24    border-top-right-radius: 5px;
25  }
26
27  .tabs label:hover {
28    background: #ccc;
29  }
30
31  .tab-content {
32    background: white;
33    padding: 20px;
34    border: 1px solid #ddd;
35    border-top: none;
36    display: none;
37    animation: fadeIn 0.5s ease;
38  }
39
```

```
index.html # style.css X
# style.css > ...
40
41
42   #tab1:checked ~ #content1,
43   #tab2:checked ~ #content2,
44   #tab3:checked ~ #content3 {
45     display: block;
46   }
47
48
49   #tab1:checked + label,
50   #tab2:checked + label,
51   #tab3:checked + label {
52     background: #fff;
53     border-bottom: 1px solid white;
54   }
55
56   @keyframes fadeIn {
57     from {opacity: 0;}
58     to {opacity: 1;}
59   }
60
```

5)OUTPUT

